CENTER FOR PURE AND APPLIED MATHEMATICS
UNIVERSITY OF CALIFORNIA, BERKELEY

COMMUNICATION COST OF [illegible]

[illegible]

DECEMBER 1988

This report was done with support from the Center for Pure and Applied Mathematics. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of the Center for Pure and Applied Mathematics or the Department of Mathematics.

(A)

Communication Cost of Sparse Cholesky Factorization on a Hypercube

by

F. Gao† and B. N. Parlett‡

ABSTRACT

The authors

~~We~~ consider the nested dissection ordering of a $k \times k$ grid of nodes and the Cholesky factorization of the associated $k^2 \times k^2$ symmetric matrix. When the factorization is computed on a hypercube machine with p processors then the communication cost (the total number of nonzero elements that need to be communicated among the processors) can be kept down to $O(pk^2)$ when the processors are assigned appropriately. This result was proved in George, Liu and Ng (1987). We offer a simpler proof of a slightly stronger result: the communication cost for each processor is $O(k^2)$. Load balancing is built into our proof. Our argument extends to grids in more than 2 dimensions. (KP)

*This work was supported in part by Contract N00014-85-K-0180 with the Office of Naval Research.

†Department of Computer Science, University of British Columbia, Vancouver, V6T 1W5, Canada, and Nankai Institute of Mathematics, Tianjin, China.

‡Department of Mathematics and Computer Science Division, University of California, Berkeley, Ca 94720.

1. Summary

We assume that the reader is familiar with the topic of solving sparse, symmetric positive definite system of equations and the (asymptotically) optimal properties of the nested dissection ordering of the unknowns. See [George and Liu, 1980] for a full discussion of the topic.

When a p-processor hypercube is available for computing the Cholesky factorization of the coefficient matrix associated with a $k \times k$ grid there arises the task of assigning the grid nodes (i.e. the associated columns of the matrix) among the p processors. This note considers the associated communication cost (i.e. the total number of nonzero values that each processor must send or receive) under the quadrant to subcube assignment strategy proposed in [George, Liu, and Ng, 1987]. There it was shown that this communication cost is $O(pk^2)$. In this note we present a simpler proof that the communication cost for each processor is $O(k^2)$.

Section 2 offers a brief discussion of nested dissection. Section 3 outlines the results of [George, Liu, and Ng, 1987] and describes their preferred method of assigning nodes to processors. Section 4 states their key technical lemmas and Section 5 presents the proof of our bound.

We recall that a symmetric positive definite matrix A admits a unique Cholesky factorization $A = LL^t$ where L is lower triangular and L^t denotes its transpose. The discretization of a 2nd order elliptic boundary value problem set on a square domain leads to a system of equations $Au = f$ with a very special, sparse, coefficient matrix A.

When the Cholesky factorization of A is implemented on a multiprocessor the information needed to compute a particular column of L will be distributed among several processors. The key question addressed here is how many nonzero values must be transmitted during the computation of L?

This brings us to the next section.

2. Nested Dissection

The primary challenge posed by sparse matrices is to find orderings that preserve sparsity during factorization. When A is associated with a $k \times k$ grid of mesh points (one 'unknown' for

each mesh point or **node**) then there is a sophisticated way to order the nodes. It is of most use when there is only 'nearest neighbor' connections between the nodes.

The Nested Dissection Ordering (ND):

> Select a +-shaped subset of nodes, called the separator, that divides the remaining nodes into 4 equal (or nearly equal) quadrants. (It helps if $k = 2^n - 1$.) Number the $2k-1$ nodes of the separator first. This gives Level 0 of the ordering. Now proceed to label the four quadrants in the same way. Their separators are labeled next, imposing some fixed order on the quadrants, say NW, NE, SW, SE. This gives Level 1. At Level 2 there are 16 subgrids whose separators must be labeled. The process terminates at a level where the subgrids are too small to be subdivided and the whole subgrid is declared to be the separator at this level.

When $k = 2^n - 1$ the 4^l subgrids at level l are $(2^{n-l} - 1) \times (2^{n-l} - 1)$. The last level is $n-1$.

In order to compute the Cholesky factorization the unknowns are eliminated in **reverse** order (largest label goes first). However the order given above is satisfactory for an analysis of communication. There is a column of L associated with each node of the $k \times k$ grid.

From the description of ND it is clear that, as the levels increase, the life cycle of a particular node comprises three stages:

1. quadrant node,
2. separator node,
3. boundary node.

If a node is in a separator at level l then, at any level $m > l$, it will be on the boundary of two or more subgrids at level m. For some nodes the first stage is skipped, for others the last.

Two properties of ND are needed in our analysis.

P1. Each node is in the separator of a unique subgrid G at some level l.

We say that a pair of nodes belong to each other if the corresponding element of L is nonzero.

P2. Let node v be in the separator of subgrid G. Then all nodes belonging to v either lie on this separator or on G's boundary.

These properties, and many others, are established using the notion of reachable sets, in the classic book [George and Liu, 1980]. One advantage of George's theory is that the analysis of the factorization is greatly simplified when translated into relationships among the nodes of the grid.

George showed that for the ND ordering of a $k \times k$ grid the number of nonzeros in L does not exceed $8k^2\log_2 k$ and the number of scalar multiplications required to compute L does not exceed $10k^3$. Recall that A is of order k^2. He also showed that for large k no ordering can produce fewer than $O(k^2\log_2 k)$ nonzeros nor take fewer than $O(k^3)$ multiplications. Thus, for scalar arithmetic, ND is asymptotically optimal.

3. The [George, Liu, and Ng, 1987] Paper

The number of nonzeros that must be passed between the p processors depends to a certain extent on the way in which the original data is distributed among them. An earlier paper, [George, Heath, Liu, and Ng, 1986] analysed a general wrap around assignment of columns to processors in which column i went to processor $(i-1) \bmod p$. Such an assignment guarantees that each processor is involved in $O(k^2\log_2 k)$ send or receive operations.

In [George, Liu, and Ng, 1987] a new assignment strategy is examined. This one corresponds in a natural way to the hierarchy induced by ND. It is called the subtree-to-subcube assignment in that paper which makes use of the concept of the elimination tree. Since we make no reference to elimination trees we give it a new name.

The Quadrant to Subcube Assignment Procedure (Q-S):

Take the "+"-shaped separator and assign its nodes among the available processors in any load balanced way (e.g. the wrap-around order). Each of the four quadrants (subgrids) created by the separator is assigned to one of four disjoint sub (hyper) cubes obtained by dividing the current subcube into four equal subcubes of 1/4-th the size.. This procedure is continued

recursively as far as it can go. In the normal case, when $p<k$, it stops when the number of processors assigned to a subgrid is less than 4. Otherwise it stops at the last level of ND when the whole subgrid is taken as the separator.

For simplicity we will assume that p is a power of 4. The numbering of processors in a hypercube makes for a natural matching to subgrids. In Fig. 1 we show the assignment of 16 processors among the nodes of an 11×11 grid.

In order to bound the number of nonzeros that have to be transmitted between processors the authors of [George, Liu, and Ng, 1987] introduce two new terms.

- **comm**(k,p) denotes the total number of nonzeros that need to be transmitted among the processors using the Q-S assignment.
- **traffic**(i,q) denotes a particular bound for the **additional** traffic among the q processors associated with an $i\times i$ subgrid over and above the traffic between these q processors and any others that happen to be associated with the boundary nodes of this $i\times i$ subgrid.

By detailed analysis they obtain recurrence relations (actually inequalities) between traffic(i,q) and traffic($i/2,i/4$). This leads to a bound on the traffic(i,q) and after further analysis, to the main result.

Theorem 1. (of [George, Liu, and Ng, 1987]) comm(k,p) $= O(pk^2)$.

More precisely, they show that comm(k,p) $< \frac{183}{4}pk^2$. Thus the communication cost associated with the quadrant to subcube assignment is better than that for the simple wrap around scheme by a factor of $\log_2 k$. However the load balancing property of this newer assignment procedure was not established in the paper we have just summarized.

The heart of their analysis, and ours, is a technical lemma concerning nested dissection that is independent of parallel processing.

4. The Key Lemma

Consider one of the $i \times i$ subgrids (of the original $k \times k$ grid) that arises in the nested dissection process.

Lemma 2. [George, Liu, and Ng, 1987] The number of nonzeros from columns (of L) associated with the $i \times i$ subgrid that are required for modifying columns associated with nodes on this subgrid's boundary is bounded by

$$\frac{341}{12} i^2 .$$

If these boundary nodes are distributed among t processors then, to be safe, this information should be sent to each of them. George et al. have made a more careful analysis that exploits the fact that not every processor needs all this information but it yields only a change of constant in the final result on comm(k, p).

Our analysis uses Lemma 2 but avoids the two auxiliary functions comm(k, p) and traffic(i, q). One other result that we need is quoted in the proof of Lemma 2 and is derived in [George and Liu, 1980].

Lemma 3. The number of nonzeros from columns associated with the 4 quadrants of an $i \times i$ subgrid that are required for modifying columns associated with the separator is bounded by

$$\frac{31}{4} i^2 .$$

Our approach is to focus on the activity of a typical processor.

5. A Bound on Communication

We define the communication cost of a processor to be the number of nonzeros it has to send or receive. More precisely, a processor has to send a nonzero if it holds a nonzero (due to the Q-S assignment) which is needed by other processors for elimination, and a processor has to receive a nonzero if it needs a nonzero which resides at another processor.

Before presenting the short proof that the communication cost for each processor is $O(k^2)$ we recapitulate Cholesky factorization from the perspective of ND and information transfer.

For the column of L associated with grid node m

a) the diagonal element must be formed,

b) the nonzeros below the diagonal must be computed.

Now (a) requires knowledge of the nonzeros in the corresponding row of L and that is accounted for by the appropriate transmission from each node that has node m in its reachable set. On the other hand (b) requires that m receive a nonzero from each node in m's reachable set.

The key property of the ND procedure (P2 in Section 2) is that the reachable set of a grid-node in the separator of a quadrant G_l is contained in the union of the separator and the boundary of G_l. Next we turn to the parallel processing aspects of the tasks.

The Q-S assignment of processors described in Section 2 has the following consequence. Each processor, at level l, is associated with a unique ($k/2^l \times k/2^l$) subgrid. Of course, more than one processor may be associated with a particular subgrid. In Fig. 1 processor 9 is associated with the 5×5 SW quadrant at level 1 and then with a 2×2 NE subgrid (near the center) at level 2. Now consider a particular node. Recall from Section 2 that some nodes that belong to a quadrant at one level may become separator nodes at the next level down. We are ready to prove our result.

Theorem 1. The communication cost of any one processor during Cholesky factorization associated with a $k\times k$ grid using the quadrant to subcube (Q-S) assignment of processors is less than $(1643/36)\,k^2$.

Proof. The computation of L proceeds according to the levels created by the nested dissection ordering (ND), from the last level back up to the first. At each level the separator nodes of all the subgrids at this level are processed. To process a node is to compute the column of L associated with that node. The communication costs are those needed to process the separator nodes.

Now consider the activity of a single processor.

1. By the Q-S assignment there is a unique subgrid G_l, at Level l, to which this processor is assigned.

2. The nonzeros needed to process the separator nodes of G_l come from two sources:

   a) the four quadrants of G_l created by the separator. (these quadrants are subgrids at Level $l+1$)

   b) the separator of G_l. (Some separator nodes are in the reachable sets of other separator nodes).

In the worst-case this processor has to send or receive (but not both) each of these nonzeros. Thus the communication cost for this processor is the sum, over all levels, of the costs in 2(a) and 2(b). Let G_l be $i \times i$.

3. An upper bound on the cost from 2(a) is given by Lemma 2 since the separator nodes of G_l are boundary nodes for its four quadrants:

$$4 \times \frac{341}{12}\left(\frac{i}{2}\right)^2 .$$

4. The cost arising from 2(b) at Level l is accounted for under 2(a) at Level $l-1$, except when $l=1$. Here is the reason. The proof in Lemma 2 assumes (generously) that all nonzeros associated with nodes in the separator of G_l are sent to every boundary node of G_l.

Since G_l is a quadrant of G_{l-1} the nonzeros considered in the previous sentence are among those in Category 2(a) at level $l-1$.

However when $l=1$ the cost for 2(b) must be taken explicitly. By Lemma 3 it is $\frac{31}{4}k^2$.

5. For each processor,

$$\text{Comm} \le \frac{31}{4}k^2 + \sum_{m=1}^{l} \frac{341}{12}\left(\frac{k^2}{4^{m-1}}\right)$$

$$< \frac{31}{4}k^2 + \frac{341}{12} \cdot \frac{4}{3}k^2$$

$$= \frac{1643}{36}k^2 .$$

Corollary.

(i) The theorem implies Theorem 1 of [George, Liu, and Ng, 1987].

(ii) The actual number of nonzeros transmitted by any one processor, when properly executed on the hypercube, is $O(k^2)$.

Proof. For (i), it suffices to note that $\frac{1643}{36}k^2p$ accounts for the total number of nonzeros that need to be communicated among the processors.

For (ii), first note that if p is small so every processor assigned to a subgrid gets at least one separator node, and therefore has to send or receive every nonzero sent from the four quadrants in the worst-case as noted in the proof of Theorem 1, then a proper limited broadcast within the corresponding subhypercube can ensure that each processor does at most a constant number of actual transmissions for every such nonzero. In the case p is large and a processor may not get any separator node in a subgrid it is assigned to, simply assume that the nonzeros sent from the four quadrants are sent from or to every processor assigned to the subgrid, regardless of their need. This does not change the bound on communication cost but ensures the validity of the above limited broadcast argument.

Note that the assumption $p < k$ which appears in the proof in [George, Liu, and Ng, 1987] is not needed here.

For grids of dimension $s > 2$ the following result holds for the Q-S assignment.

Theorem 2. The number of nonzeros that any processor has to send or receive during Cholesky factorization associated with a $k \times k$ grid using the quadrant to subcube assignment of processors is

$$O(k^{2(s-1)}).$$

Fig. 1.

0	0	0	1	1	0	4	4	4	5	5
0	0	1	1	1	1	4	4	5	5	5
1	2	2	3	0	2	5	6	6	7	4
2	2	3	3	3	3	6	6	7	7	7
2	2	0	3	3	4	6	6	4	7	7
11	12	13	14	15	5	0	1	2	3	4
8	8	8	9	9	6	12	12	12	13	13
8	8	9	9	9	7	12	12	13	13	13
9	10	10	11	8	8	13	14	14	15	12
10	10	11	11	11	9	14	14	15	15	15
10	10	8	11	11	10	14	14	12	15	15

The quadrant to subcube assignment

11×11 grid, 16 processors.

Reference

J. A. George and J. W-H Liu, *Computer Solution of Large Sparse Positive Definite Systems,* Prentice Hall Inc., Englewood Cliffs, New Jersey (1981).

J. A. George, M. T. Heath, J. W-H Liu, and E. G-Y Ng, Sparse Cholesky factorization on a local-memory multiprocessor, Tech. Report, CS-86-02, Department of Computer Science, University of Waterloo, (1986). To appear.

J. A. George, J. W-H Liu, and E. G-Y Ng, Communication results for parallel sparse Cholesky factorization on a hypercube, Tech. Report CS-87-03, York University, (1987). To appear in *Journal of Parallel Computation.*

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) University of California, Berkeley	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

Communication Cost of Sparse Cholesky Factorization on a Hypercube

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

CPAM report, December 1988

5. AUTHOR(S) (First name, middle initial, last name)

F. Gao and B. N. Parlett

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO OF REFS
December 1988	10	

8a. CONTRACT OR GRANT NO. N00014-85-K-0180	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Mathematics Branch Office of Naval Research Washington, DC 20360

13. ABSTRACT

We consider the nested dissection ordering of a $k \times k$ grid of nodes and the Cholesky factorization of the associated $k^2 \times k^2$ symmetric matrix. When the factorization is computed on a hypercube machine with p processors then the communication cost (the total number of nonzero elements that need to be communicated among the processors) can be kept down to $O(pk^2)$ when the processors are assigned appropriately. This result was proved in George, Liu and Ng (1987). We offer a simpler proof of a slightly stronger result: the communication cost for each processor is $O(k^2)$. Load balancing is built into our proof. Our argument extends to grids in more than 2 dimensions.